

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平8-297643

(43)公開日 平成8年(1996)11月12日

(51)Int.Cl. <sup>6</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 15/16	3 8 0		G 0 6 F 15/16	3 8 0
9/46	3 4 0		9/46	3 4 0 F
	3 6 0			3 6 0 C

審査請求 未請求 請求項の数 1 O L (全 4 頁)

(21)出願番号 特願平7-101934

(22)出願日 平成7年(1995)4月26日

(71)出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72)発明者 目黒 恭司

東京都大田区大森北三丁目2番16号日立シ

ステムエンジニアリング株式会社内

(74)代理人 弁理士 小川 勝男

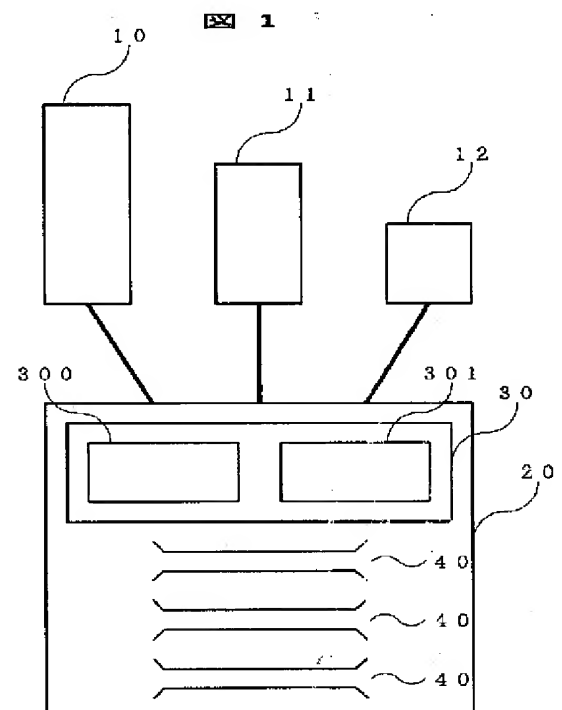
(54)【発明の名称】 複数CPUでの処理分散方式

(57)【要約】

【目的】複数CPUをクラスタ接続し各CPU毎の利用率をもとに負荷分散する方法において、各CPUで同一のハード／ソフト環境を常時用意する必要をなくし、CPU間での高度な排他処理によるオーバーヘッドを削減する。

【構成】処理能力の異なるCPUを複数台クラスタ接続し分散処理を行うシステムにおいて、同一の処理手順をふくむ処理要求は分散せず特定CPUで処理を行い、手順の異なる処理は各CPUで分散処理を行う。また、ある一定期間（時間単位、日単位等）で処理手順毎の処理回数が大きく変動する場合、その都度適切な能力のCPUへ処理を変更する。

【効果】同一のハード／ソフト環境を用意する必要はなくなりリソースの有効活用が図れる。処理発生頻度に見合ったCPU能力を得ることができる。



## 【特許請求の範囲】

【請求項1】処理能力の異なるCPUを複数台クラスタ接続し、複数処理要求の負荷を各CPUへ分散して処理するシステムにおいて、同一の処理手順を踏む処理要求毎の処理回数をもとに各CPUへ負荷分散を行う処理方式。

## 【発明の詳細な説明】

【産業上の利用分野】処理能力の異なるCPUを複数台クラスタ接続し、複数処理要求の負荷を各CPUへ分散して処理するシステムでの負荷分散方式に関する。

【従来の技術】従来の複数CPUでの負荷分散方式では、各CPU毎の演算部／メモリ／入出力制御部等ハードウェア資源の利用率をもとに負荷を分散する方法が主流である。

【発明が解決しようとする課題】複数CPUをクラスタ接続し各CPU毎の利用率をもとに負荷を分散する方法では、処理要求と当該処理要求を処理するCPUとの対応は不特定であるため、全CPUで同一のハード／ソフト環境を常時用意しておき各CPU間での高度な排他処理を必要とし、この排他処理によるオーバーヘッドが大きな課題となっていた。本発明の目的は、上記課題を解決し、同一のハード／ソフト環境を全てのCPUに用意する必要なくリソースの有効利用を図れる複数CPUでの処理分散方式を提供することである。

【課題を解決するための手段】同一の処理手順を含む処理要求については分散せず特定のCPUで処理させることが望ましい。また、ある一定期間（時間単位、日単位等）で処理手順毎の処理回数が大きく変動するようなシステムでは、その都度適切な能力のCPUへ処理を変更していく方法が望ましい。そこで本発明では、前記目的を達成するために、処理能力の異なるCPUを複数台クラスタ接続し分散処理を行うシステムにおいて、同一の処理手順をふむ処理要求は分散せず特定CPUで処理を行うが、手順の異なる処理は各CPUで分散処理を行う。また、本発明の処理能力の異なるCPUを複数台クラスタ接続し処理の負荷分散を行う方式は、同一の処理手順を踏む処理要求毎に処理回数の統計情報を取り、その処理回数をもとに、処理要求とその処理要求を処理するCPUとを対応づけることにする。

【作用】複数の処理要求を無差別に処理分散するのではなく、同一の処理手順をふむ処理要求は分散せず特定CPUで処理を行うが、手順の異なる処理は各CPUで分散処理を行うことにより、同一のハード／ソフト環境を用意する必要はなくなり必要な処理手順のリソース分で済みリソースの有効活用が図れる。これにより、プログラムの常駐などが容易になり処理速度向上に効果がある。また、その処理回数の変動で処理するCPUを変更することにより、処理発生頻度に見合ったCPU能力を得ることができる。

【実施例】本実施例では処理要求を各々の処理手順が異

なり同等の処理能力を必要とするA～Fのトランザクションに特定し、日々、各トランザクションの処理回数が大きく変動するため必要とするCPU能力を定期的に見直すシステムを想定する。以下図面を参照して説明する。図1は本発明の一実施例を示すブロック図、図2は各トランザクションの統計情報テーブル、図3は各トランザクションと処理させるCPUを対応づける処理分散管理テーブル、図4は本実施例の動作を示すフローチャートである。図1に示すように、処理能力の異なるCPU10、11、12と処理分散制御部30を有し、各CPUと接続される結合装置20とからなる。処理分散制御部30には、トランザクションの処理回数をカウントする統計情報カウンタ300と、トランザクションと処理CPUの対応づけを設定する処理分散管理テーブル301と分散されたトランザクションを各CPU毎に一時蓄積する処理キュー40とが含まれる。次に動作について説明する。各トランザクションには業務種別毎に識別子としてA～Fのトランザクションコードが付加されており、初期設定として図2に示すように統計情報カウンタ300の前日統計に従い、処理分散管理テーブル301へ50回以上のトランザクションA／B／CはCPU1で処理、10回以上50回未満のトランザクションD／EはCPU2で処理、10回未満のトランザクションFはCPU3で処理と設定するものとする。また、各CPUでは処理能力に応じて処理可能なトランザクションコードの数をCPU1は4、CPU2は3、CPU3は2と設定する。トランザクションA～Fが各CPU経由で結合装置20へ入力されると、処理分散制御部30は統計情報カウンタ300にてトランザクションA～F毎に受け付け回数の統計をとる501。次に、処理分散制御部30は統計情報カウンタ300内の当日の処理回数と処理分散管理テーブル301内の分散条件との比較を行う502。比較の結果、当日の処理回数が分散条件の上限値に満たない場合（トランザクションD以外は分散条件の上限値に満たない）、次のトランザクションをカウントする。上限値を超えた場合（トランザクションDの場合）、処理分散制御部30は超えた値（トランザクションD：60回）が適合する分散条件（50回以上）に設定されているトランザクション（トランザクションA／B／C）の当日の処理回数を確認する503。設定されているトランザクション（トランザクションA／B／C）の当日の処理回数が分散条件の下限値を下回る場合（トランザクションAが下回っている）、処理分散制御部30は処理分散管理テーブル301内の上限値を超えたトランザクション（トランザクションD）と下限値を下回っているトランザクション（トランザクションA）との設定を入れ替える504。また、分散条件の下限値を下回るトランザクションがない場合は、そのCPUで処理可能なトランザクションコードの数を確認する506。処理分散制御部30は、処理分散管理テーブル

301内のCPUに設定されているトランザクションコードが設定値(トランザクションコードの数)に満たない場合、上限値を超えたトランザクションを適合する分散条件のCPUへ設定変更を行う507。トランザクションコードが設定値(トランザクションコードの数)に達している場合、次のトランザクションをカウントする。次に処理分散制御部30は受け付けたトランザクションを処理分散管理テーブル301の設定に従い処理すべきCPUのキューへ分散する。各CPUは自分のキューからトランザクションを取り出し処理を行う。本実施例では、各トランザクションは同等の処理能力を必要とすると仮定したため、処理回数そのものが処理CPUの分散条件となっているが、各トランザクションの必要とする処理能力が異なる場合は、各トランザクションの処理の重みと処理回数を考慮した分散条件を設定すれば同様のことが行える。

【発明の効果】本発明によれば、同一の処理手順をふむ処理要求は分散せず特定CPUで処理を行うが、手順の

異なる処理は各CPUで分散処理を行うことにより、同一のハード/ソフト環境を用意する必要はなくなりリソースの有効活用が図れる。また、その処理回数の変動で処理するCPUを変更することにより、処理発生の頻度に見合ったCPU能力を得ることができる。

【図面の簡単な説明】

【図1】処理能力の異なるCPUを複数台クラスタ接続しての処理分散方式の一実施例を示すシステム概略構成図である。

10 【図2】各処理要求の統計情報テーブルである。

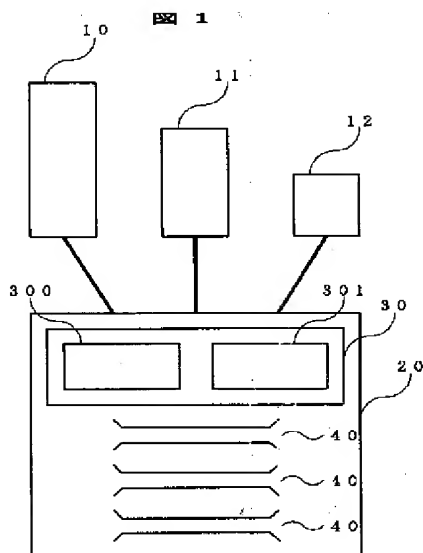
【図3】各処理要求と処理させるCPUを対応づける処理分散管理テーブルである。

【図4】処理分散制御部による処理要求と処理するCPUの対応づけ変更のフローチャートである。

【符号の説明】

10～12…処理プロセッサ、20…結合装置、30…処理分散制御部、40…処理キュー、300…統計情報カウンタ、301…処理分散管理テーブル。

【図1】



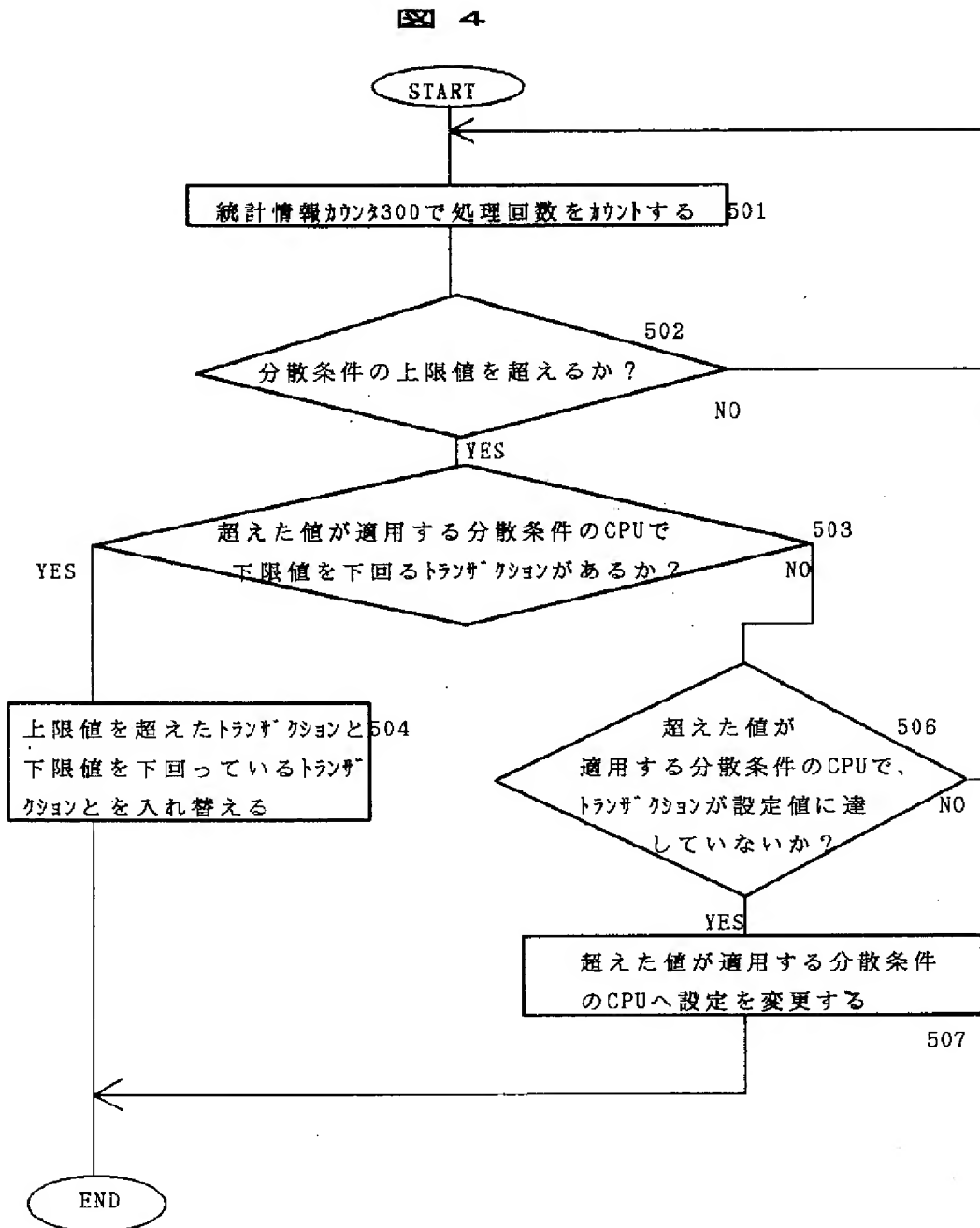
【図2】

トランザクション コード	統計 (回数)			
	前日	当日	先月	当月
A	51	20	100	100
B	60	70	150	200
C	150	80	3000	1000
D	20	60	400	200
E	15	10	300	50
F	2	0	20	4

【図3】

CPU No.	1	2	3
分散条件	50回以上	50回未満10回以上	10回未満
トランザクション コードの数	4	3	2
トランザクション コード	A B C	D E	F

【図4】



**DERWENT-ACC-NO:** 1997-039710

**DERWENT-WEEK:** 199704

*COPYRIGHT 2008 DERWENT INFORMATION LTD*

**TITLE:** Load distribution method for multi processor system involves providing distinct processing loads to several CPUs with different processing capabilities based on processing frequency of processing demand needed to complete processing routine

**INVENTOR:** MEGURO K

**PATENT-ASSIGNEE:** HITACHI LTD[HITA]

**PRIORITY-DATA:** 1995JP-101934 (April 26, 1995)

**PATENT-FAMILY:**

<b>PUB-NO</b>	<b>PUB-DATE</b>	<b>LANGUAGE</b>
JP 08297643 A	November 12, 1996	JA

**APPLICATION-DATA:**

<b>PUB-NO</b>	<b>APPL- DESCRIPTOR</b>	<b>APPL- NO</b>	<b>APPL- DATE</b>
JP 08297643A	N/A	1995JP- 101934	April 26, 1995

**INT-CL-CURRENT:**

<b>TYPE</b>	<b>IPC DATE</b>
CIPP	G06F15/16 20060101
CIPS	G06F9/46 20060101
CIPS	G06F9/50 20060101

**ABSTRACTED-PUB-NO:** JP 08297643 A**BASIC-ABSTRACT:**

The method involves distributing load in each of the several central processing units (10-12) with different processing capabilities that are connected in a cluster . The load is distributed according to the type and frequency requirement of the process which prevent any identical process be given to the same CPU. If the load requires a faster frequency of processing, the CPU suited to the process and is changed in constant periods.

**ADVANTAGE** - Improves resources utilisation by

distributing load according to processing capability of CPU.

**CHOSEN-DRAWING:** Dwg.1/4

**TITLE-TERMS:** LOAD DISTRIBUTE  
METHOD MULTI  
PROCESSOR SYSTEM  
DISTINCT PROCESS CPU  
CAPABLE BASED  
FREQUENCY DEMAND  
NEED COMPLETE  
ROUTINE

**ADDL-INDEXING-TERMS:** CPU

**DERWENT-CLASS:** T01

**EPI-CODES:** T01-F02C2;

**SECONDARY-ACC-NO:**

**Non-CPI Secondary Accession Numbers:** 1997-033196